

①⑨ RÉPUBLIQUE FRANÇAISE
INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE
PARIS

①① N° de publication :
(à n'utiliser que pour les
commandes de reproduction)

2 716 058

②① N° d'enregistrement national : **94 01271**

⑤① Int Cl⁶ : H 04 L 9/30

⑫

DEMANDE DE BREVET D'INVENTION

A1

⑫② Date de dépôt : 04.02.94.

⑫③ Priorité :

⑫④ Date de la mise à disposition du public de la
demande : 11.08.95 Bulletin 95/32.

⑫⑤ Liste des documents cités dans le rapport de
recherche préliminaire : *Se reporter à la fin du
présent fascicule.*

⑫⑥ Références à d'autres documents nationaux
apparentés :

⑦① Demandeur(s) : *FRANCE TELECOM, Etablissement
autonome de droit public — FR et LA POSTE — FR.*

⑦② Inventeur(s) : Girault Marc.

⑦③ Titulaire(s) :

⑦④ Mandataire : Société de Protection des Inventions.

⑫④ Procédé de signature numérique et d'authentification de messages utilisant un logarithme discret.

⑫⑦ Selon l'invention, on réduit le nombre de multiplica-
tions modulaires à effectuer. Dans une variante, ces multi-
plications modulaires sont même supprimées et l'entité qui
signe utilise des nombres précalculés.
Application en télécommunications.

FR 2 716 058 - A1



PROCEDE DE SIGNATURE NUMERIQUE ET D'AUTHENTIFICATION
DE MESSAGES UTILISANT UN LOGARITHME DISCRET

DESCRIPTION

5

Domaine technique

La présente invention a pour objet un procédé de
signature numérique et d'authentification de messages
10 utilisant un logarithme discret. Elle trouve une
application dans la cryptographie dite à clé publique.

L'invention concerne plus particulièrement le
domaine des procédés dit "sans transfert de
connaissance" (ou "zero-knowledge"). Dans de tels
15 procédés, l'authentification se déroule suivant un
protocole qui, de façon prouvée, et sous des hypothèses
reconnues comme parfaitement raisonnables par la
communauté scientifique, ne révèle rien de la clé
secrète de l'entité dont la signature est à
20 authentifier.

Etat de la technique antérieure

Dans certaines techniques antérieures de signature
25 numérique de messages, l'entité devant émettre un
message signé est détentrice d'une clé dont un élément,
appelé module, est public et un autre élément est
secret. Pour signer un message, cette entité ajoute au
message une redondance en utilisant sa clé secrète. Le
30 message émis est donc une fonction du message et de la
clé secrète.

Pour vérifier la signature, le destinataire du
message utilise la partie publique de la clé et
effectue un calcul qui doit redonner le message initial
35 avec sa redondance. En retrouvant la redondance, le

destinataire conclut que le message qu'il a reçu n'a pu être transmis que par l'entité qui prétend l'avoir envoyé, puisqu'elle seule était capable de traiter le message de cette manière.

5

Dans de telles procédures, la sécurité est fondée sur l'extrême difficulté qu'il y a à inverser certaines fonctions. Dans le cas de l'invention qui va être décrite, la sécurité du procédé de signature est fondée
10 sur l'extrême difficulté du problème du logarithme discret. Ce problème consiste, étant donné la relation mathématique : $y = g^x \text{ modulo } n$ (qui signifie : y est le reste de la division de g^x par n), à retrouver x lorsque l'on connaît n , g et y . Ce problème est
15 impossible à résoudre, en l'état actuel des connaissances, dès que la taille n atteint ou dépasse 500 bits et que celle de x atteint ou dépasse 128 bits.

Dans les systèmes mettant en oeuvre de tels procédés, il existe en général une autorité, en
20 laquelle un ensemble d'utilisateurs a confiance, qui détermine un nombre n de grande taille (au moins 500 bits), constituant le module. L'invention s'applique dès lors que le nombre n est choisi de telle sorte que le problème du logarithme discret soit impossible à
25 résoudre en pratique. Pour cela, il existe essentiellement deux options :

- ou le module n est calculé de telle sorte qu'il soit impossible, en pratique, (c'est-à-dire compte tenu des algorithmes de factorisation existants), de retrouver les facteurs premiers
30 dont il est le produit ; dans une utilisation classique de cette option, n est le produit de deux grands facteurs premiers distincts ;
- ou ce module est un nombre premier vérifiant
35 certaines propriétés que l'on peut trouver dans

la littérature spécialisée ; dans une utilisation classique de cette option, $n-1$ est le double d'un nombre premier.

L'autorité choisit ensuite un entier g , appelé
5 base, tel que l'ensemble engendré par g , c'est-à-dire l'ensemble constitué des nombres g^x modulo n , lorsque x parcourt l'intervalle $[0, n-1]$, soit un sous-ensemble de taille maximale (ou, en tout cas, de taille suffisamment grande (au moins 2^{128})), de cet
10 intervalle. A titre d'exemple, la taille maximale est $n-1$ lorsque n est premier, proche de $n/2$ lorsque n est le produit de deux facteurs premiers distincts.

Les nombres n et g sont publiés par l'autorité et doivent être connus de tous les utilisateurs qui y sont
15 rattachés. Dans une variante, ces paramètres sont choisis individuellement par chaque utilisateur et font donc partie intégrante de sa clé publique.

Dans un tel contexte, on connaît un procédé de
20 signature de messages qui comprend essentiellement les opérations suivantes. Dans ce procédé, n est un nombre premier et q est un nombre premier divisant $n-1$. La clé privée x est choisie au hasard entre 1 et q et la clé publique y est prise égale à $y=g^x$ modulo n . L'entité
25 qui désire émettre et signer un message M choisit au hasard un entier positif k et calcule $r=g^k$ modulo n . Cette entité calcule ensuite une fonction de hachage (ou de condensation) $c=h(r, M)$, la fonction de hachage étant connue de tous. L'entité calcule ensuite une
30 somme $s=k+cx$ modulo n et elle transmet cette somme à une entité authenticatrice ainsi que la fonction c . Cette entité calcule, à l'aide de la clé publique et de la base g , le produit modulo n de $g^s.y^c$ soit r et vérifie que la fonction de hachage $h(r, M)$ redonne bien
35 la fonction de hachage c qu'elle a reçue.

Un tel procédé est décrit dans l'article de C.P. SCHNORR intitulé "Efficient Signature Generation by Smard Cards", publié dans J. Cryptology (1991) 4 : 161-174, pp. 161-174.

Le brevet américain US-A-4,995,082 délivré à C.P. SCHNORR décrit également un procédé analogue avec des algorithmes encore plus complexes où la somme s n'est plus seulement la somme de deux termes modulo q mais une somme double, toujours modulo un certain entier, en l'occurrence $p-1$.

Si ce procédé donne satisfaction à certains égards, il présente néanmoins l'inconvénient de nécessiter des opérations qui sont des multiplications modulaires. Ces opérations apparaissent dans le calcul de g^k modulo p et dans la somme $k+cx$ modulo q (ou les doubles sommes modulo $p-1$).

Or, cette opération de multiplication modulaire est complexe et nécessite des moyens importants. Dans des dispositifs simples n'utilisant que des microprocesseurs standards, cette opération n'est guère possible. De même, dans des ordinateurs où les fonctions cryptographiques sont réalisées à l'aide d'un logiciel, le calcul d'une multiplication modulaire n'est jamais très performant.

La réduction du nombre de tels calculs est donc souhaitable. C'est précisément le but de la présente invention.

30

Exposé de l'invention

A cette fin, la présente invention prévoit un procédé de signature de messages, du genre de celui qui

vient d'être décrit, mais qui réduit, voire supprime, les multiplications modulaires.

La réduction est obtenue en faisant calculer à l'entité émettrice une somme simple, soit $s=k+cx$, de l'entier k choisit au hasard et du produit de la fonction de hachage c et de la clé secrète x , et non plus une somme modulo un entier, comme dans l'art antérieur. Cette multiplication modulaire est donc supprimée.

10

Dans un mode de mise en oeuvre encore plus simple, les nombres r de la forme g^k modulo n , dont le calcul met en oeuvre des multiplications modulaires, sont précalculés et mémorisés dans l'entité signataire, ce qui dispense celle-ci d'avoir à les calculer. Il ne reste plus alors aucune multiplication modulaire à effectuer dans l'entité émettrice.

15

De façon précise, la présente invention a pour objet un procédé de signature numérique de message, dans lequel une autorité choisit un nombre n , appelé module, un entier g appelé base, trois paramètres t , u , v appelés paramètres de sécurité, cette autorité publiant le module n , la base g et les trois paramètres de sécurité t , u , v auprès de diverses entités utilisatrices ; ce procédé comprenant, pour une entité A désirant émettre un message M et le signer numériquement, et pour une entité B désirant vérifier la signature du message M , les opérations suivantes :

25

a) l'entité A choisit au hasard une clé secrète x égale à un entier positif ou nul strictement inférieur au paramètre t et détermine sa clé publique en élevant g à une certaine puissance de x modulo n et cette clé publique est certifiée par l'autorité,

30

35

- b) l'entité A choisit ensuite, au hasard, un entier positif ou nul k strictement inférieur au produit des trois paramètres t , u , v et l'entité A garde l'entier k secret,
- 5 c) l'entité A calcule ensuite, à l'aide de g et de k , un entier r tel que :
- $$r = g^k \text{ modulo } n,$$
- d) l'entité A calcule ensuite, par une fonction de hachage connue de toutes les entités utilisatrices, la fonction de hachage de l'entier r précédemment obtenu et du message à signer M , soit $c=h(r, M)$, c étant un entier compris entre 0 et $v-1$,
- 10 e) l'entité A calcule, à partir de sa clé secrète x , de l'entier k et de la fonction de hachage c , un entier s défini par :
- $$s = k + cx$$
- 15 f) l'entité A transmet le message M , la fonction de hachage c et l'entier s précédemment obtenu à l'entité B,
- 20 g) l'entité B vérifie le certificat de la clé publique de A et calcule, à partir du module n , de la clé publique y de l'entité A et de la fonction de hachage c reçue de A, le produit modulo n , de y^c par g^s ,
- 25 soit $r = y^c g^s \text{ modulo } n$
- h) l'entité B vérifie alors que la fonction de hachage du produit r qu'elle a obtenu et du message M qu'elle a reçu, soit $h(r, M)$, redonne bien la fonction de hachage c reçue de A, auquel cas l'authentification de la signature du message émis par A est réussie.
- 30

Dans une variante, r est remplacé par un condensé de r , par exemple $h(r)$ ou les 128 bits de poids faible (ou poids fort) de r .

De préférence, l'entité A possède un ensemble
5 prédéterminé de valeurs $k_1, \dots, k_i, \dots, k_m$ et un ensemble correspondant d'entiers $r_1, \dots, r_i, \dots, r_m$ liés aux k_i par $r_i = g^{k_i}$ modulo n . Dans ce cas, l'entité A , pour déterminer ledit entier r , choisit un nombre k_i dans l'ensemble des k_i et lit l'entier correspondant
10 $r_i = g^{k_i}$ modulo n .

De préférence encore, l'autorité et l'entité A possèdent deux générateurs pseudo-aléatoires identiques aptes à délivrer tous deux la même suite pseudo-aléatoire d'entiers $k_1, \dots, k_i, \dots, k_m$ à partir d'une
15 valeur d'initialisation k_0 appliqué au générateur. Cette valeur d'initialisation k_0 est échangée entre l'entité A et l'autorité de manière sécurisée. L'autorité calcule alors la séquence d'entiers $r_i = g^{k_i}$ modulo n correspondant à la séquence $k_1, \dots, k_i, \dots,$
20 k_m liée à l'initialisation k_0 , transmet la séquence des entiers $r_1, \dots, r_i, \dots, r_m$ à l'entité A qui les mémorise. L'entité A devant émettre et signer des messages $M_1, \dots, M_i, \dots, M_m$, met en route son générateur pseudo-aléatoire à partir de la même valeur
25 d'initialisation k_0 , et, pour chaque message M_i à signer, lit la valeur k_i délivrée par son générateur pseudo-aléatoire, lit l'entier r_i correspondant et signe le message M_i à l'aide de l'entier r_i .

La présente invention a également pour objet un
30 procédé d'authentification de message qui peut être considéré comme une variante du procédé de signature.

Exposé détaillé de modes de réalisation

Le procédé de l'invention est précédé d'une phase de prétraitement, dans laquelle l'autorité choisit un
5 module n de grande taille, c'est-à-dire d'au moins 500 bits et une base g . L'autorité détermine également la valeur des trois paramètres de sécurité, t , u et v . Il est recommandé que la longueur de t soit d'au moins 128 bits, celle de u d'au moins 64 bits et celle de v d'au
10 moins 128 bits.

L'entité utilisatrice, que l'on désignera par A , choisit au hasard une clé secrète x , qui est un entier positif ou nul strictement inférieur au paramètre t . L'entité A calcule sa clé publique en
15 élevant g à une certaine puissance de x modulo n , par exemple $y = g^{-x}$ modulo n ou $y = g^x$ modulo n ou par d'autres équations proches de celle-ci. L'inverse se calcule à l'aide de l'algorithme d'EUCLIDE de calcul du plus grand commun diviseur.

20 La clé publique y (ou n , g et y) est ensuite certifiée par l'autorité, selon un mécanisme de certification quelconque. Si ce mécanisme repose sur un procédé de signature numérique tel que celui qui va être décrit, l'autorité calcule un certificat, c'est-à-
25 dire une signature numérique d'un ensemble de paramètres contenant au minimum la clé publique y (ou n , g et y) et une chaîne I qui rassemble des caractéristiques discriminantes de l'utilisateur associé. Par exemple, I peut être la concaténation,
30 selon un format à définir avec précision, de l'identité de l'utilisateur, de son adresse, de son âge, etc. ainsi que, par exemple, de ses droits à effectuer telle ou telle action.

Tous les utilisateurs devront connaître la clé
35 publique de l'autorité, afin de pouvoir vérifier les

certificats qu'elle émet, et connaître ainsi avec certitude la clé publique correspondant à l'utilisateur dont le champ d'identification est I. Ce type de procédure fait partie de l'état de la technique en
5 matière de cryptographie.

Cette phase préliminaire étant achevée, le procédé de signature de l'invention comprend certaines opérations qui vont être décrites maintenant. La clé
10 publique de A est supposée connue de B, ce qui peut résulter d'un échange préalable non explicité ici, qui pourrait d'ailleurs être intégré dans le protocole de signature.

1) L'entité A choisit au hasard un entier k positif ou nul strictement inférieur au produit $t.u.v=2^{400}$ et garde secret cet entier k. Elle calcule ensuite l'entier $r=g^k$ modulo n.

2) L'entité A calcule ensuite $c=h(r, M)$ où h est une fonction de hachage.

20 3) L'entité A calcule alors l'entier $s=k+cs$.

4) L'entité A envoie (M, c, s) à l'entité B.

5) L'entité B calcule le produit, modulo n, de y^c par g^s :

$$r = y^c g^s \text{ modulo } n$$

25 et vérifie que

$$h(r, M) = c$$

Si la vérification est faite, l'authentification de la signature M est réussie.

30 On remarquera que la signature du message a, pour les valeurs de paramètres prises en exemple, une longueur de 560 bits. Avec des valeurs un peu inférieures, mais ne compromettant pas la sécurité du procédé, on pourrait obtenir une longueur d'environ 450
35 bits.

Dans un mode de réalisation particulier qui va maintenant être décrit, l'entité A n'aura qu'une seule opération à effectuer : celle de l'étape 3. Aucune opération modulaire ne sera donc à mettre en oeuvre.

5 A cette fin, l'entité A peut être équipée d'une table (ou mémoire) où les entiers $r=g^k$ sont mémorisés à l'adresse k. Il suffit alors à l'entité A de venir lire le nombre r correspondant au nombre k choisi.

10 En contrepartie, ce dispositif ne peut effectuer qu'un nombre limité de signatures. Cependant, lorsque ce nombre est atteint, on peut, par une procédure de rechargement (éventuellement à distance), mettre à nouveau à la disposition de A un nouveau jeu d'entiers r.

15 Dans un mode particulier de réalisation, les nombres k utilisés pour les signatures successives sont engendrés par un générateur pseudo-aléatoire partagé par l'utilisateur et l'autorité. La valeur initiale de
20 ce générateur doit être transmise par l'autorité à l'utilisateur (ou le contraire) de façon sécurisée. Cette sécurité peut être obtenue en effectuant cette transmission localement, ou en effectuant cette transmission à distance à l'aide d'un procédé
25 cryptographique (par exemple un procédé de chiffrement) indépendant du procédé actuellement décrit.

 Soit k_0 cette valeur initiale. Soient $k_1, \dots, k_i, \dots, k_m$ les valeurs engendrées par le générateur pseudo-aléatoire, l'indice i étant l'indice courant
30 allant de 1 à m. L'autorité calcule les valeurs $r_1, \dots, r_i, \dots, r_m$ correspondantes (par la formule de l'étape 1) et les transmet à l'utilisateur (sans qu'il y ait besoin d'en assurer la confidentialité). L'utilisateur les mémorise. Avantageusement, l'autorité
35 ne transmettra qu'un condensé de chaque valeur $r_1, \dots,$

r_i, \dots, r_m (par exemple les 160 derniers bits), et le calcul de l'étape 2 ou 5 ne portera que sur ces valeurs tronquées.

L'étape 1 du procédé de signature est alors
5 modifiée comme suit :

1) L'entité A met en oeuvre son générateur pseudo-aléatoire pour engendrer k , une valeur strictement inférieure au produit $tuv=2^{400}$, et va chercher en mémoire la valeur de r (ou la valeur tronquée de r)
10 correspondante que l'autorité lui avait préalablement transmise.

Lorsque l'entité A a produit ses m signatures, elle redemande alors à l'autorité une nouvelle initialisation $k'0$ (transmis de façon sûre) et une
15 nouvelle séquence $r'1, \dots, r'i, \dots, r'm$. L'entité A peut signer à nouveau m messages à l'aide des $r'1, \dots, r'i, \dots, r'm$.

A titre d'exemple, si chaque nombre r_i fait 160 bits, alors, une mémoire de 2 kilooctets permet de
20 produire environ 100 signatures.

La production de telles signatures est quasiment instantanée, même avec un dispositif muni de ressources limitées comme une carte à microprocesseur standard.

25 Deux variantes permettent de réduire la taille de la signature.

Dans la première, après avoir calculé s , l'entité A vérifie que $c(t-1) \leq s \leq tuv-1$. Si ce n'est pas le cas, A (et B dans le cas de l'authentification de message)
30 recommence le processus de signature (ou d'authentification de message) à l'étape b), c'est-à-dire : l'entité A choisit au hasard un entier k etc.... On est ainsi certain que s ne révèle aucune information sur le secret x . Ceci permet de réduire u autant qu'on
35 le désire, mais au prix de reprises éventuelles.

Dans la seconde variante, on remplace dans l'équation $c=h(r,M)$, M par $f(M)$ où f est également une fonction de hachage. Dans le cas où les étapes b) à e) sont exécutées par un dispositif sécurisé, les qualités
5 requises pour h sont moindres : il suffit que h soit à sens unique. Ceci permet de diviser environ par deux la longueur de v .

Dans la description qui précède, l'accent a été
10 mis sur la signature de messages mais l'invention s'applique également à l'authentification de messages qui en est une variante. La différence entre l'authentification de message et la signature numérique de message est que la première résulte d'un procédé
15 interactif et la seconde d'un procédé non interactif. Une conséquence est que les données servant à authentifier un message venant d'une entité A auprès d'une entité B ne peuvent pas être utilisées ultérieurement pour authentifier le même message auprès
20 d'une troisième entité C. Au contraire, une signature numérique peut être vérifiée par n'importe quelle entité à n'importe quel moment.

Dans la variante d'authentification, les opérations sont les suivantes :

25 1) L'entité A choisit au hasard un entier positif ou nul k strictement inférieur au produit $tuv=2^{270}$ et elle garde k secret. Elle calcule ensuite l'entier :

$$r = g^k \text{ (modulo } n)$$

puis

30 $r' = h(r,M)$

2) L'entité A envoie r' à l'entité B.

3) L'entité B choisit au hasard un entier positif ou nul c strictement inférieur à $v = 2^{30}$ (c'est-à-dire inférieur à $v-1$).

35 4) L'entité B envoie c à l'entité A.

5) L'entité A calcule l'entier :

$$k = k + cx.$$

6) L'entité A envoie (M,s) à l'entité B.

7) L'entité B calcule le produit modulo n de y^C
5 par g^S :

$$y^C g^S = r \text{ (modulo } n)$$

et vérifie que

$$h(r,M) = r'.$$

Si la vérification est satisfaite,
10 l'authentification de l'entité A et du message M est
réussie.

Les dispositifs et moyens permettant de mettre en
oeuvre le procédé de signature qui vient d'être décrit
15 sont classiques. On peut simplement souligner que les
entités doivent :

- être rattachées à une autorité de tutelle, qui
calcule les certificats, produit les nombres n et
g (à moins que chaque utilisateur ne les produise
20 lui-même) et fixe les valeurs des paramètres de
sécurité ;
- être capables de détenir les clés secrètes (pour
l'entité à authentifier), ce qui suppose un moyen
de stockage sécurisé, de façon à ce qu'une
25 lecture non autorisée soit impossible en
pratique ;
- être capables de détenir des clés publiques, ce
qui suppose un moyen de stockage sécurisé, de
façon à ce qu'une modification non autorisée soit
30 impossible en pratique ;
- être capables, dans la variante générale,
d'effectuer des multiplications modulo n, ce qui
suppose (dans le cas d'une carte à
microprocesseur), l'existence d'une unité de
35 calcul spécialisée ; être capables, dans le mode

de réalisation particulier, d'effectuer des opérations arithmétiques usuelles, ce qui évite (dans le cas d'une carte à microprocesseur), l'existence d'une unité de calcul spécialisée ;

- 5 - être capables de mettre en oeuvre un générateur aléatoire et/ou un générateur pseudo-aléatoire ;
- être capables de dialoguer, ce qui suppose l'existence d'une interface de communication.

10 Tous ces moyens sont connus de l'homme du métier.

REVENDICATIONS

1. Procédé de signature numérique de message dans lequel une autorité choisit un nombre n , appelé module, un entier g , appelé base, trois paramètres t , u , v , appelés paramètres de sécurité, cette autorité publiant le module n , la base g et les trois paramètres de sécurité t , u , v auprès de diverses entités utilisatrices, ce procédé comprenant, pour une entité A désirant émettre un message M et le signer numériquement, et pour une entité B désirant vérifier la signature du message M , les opérations suivantes :

- a) l'entité A choisit au hasard une clé secrète x égale à un entier positif ou nul strictement inférieur au paramètre t et détermine sa clé publique en élevant g à une certaine puissance de x modulo n et cette clé publique est certifiée par l'autorité,
- b) l'entité A choisit ensuite, au hasard, un entier positif ou nul k , strictement inférieur au produit des trois paramètres t , u , v et l'entité A garde l'entier k secret,
- c) l'entité A calcule ensuite, à l'aide de g et k , un entier r tel que :
$$r = g^k \text{ modulo } n,$$
- d) l'entité A calcule ensuite, par une fonction de hachage connue de toutes les entités utilisatrices, la fonction de hachage de l'entier r précédemment obtenu et du message à signer M , soit $c=h(r, M)$, c étant un entier compris entre 0 et $v-1$,
- e) l'entité A calcule, à partir de sa clé secrète x , de l'entier k et de la fonction de hachage c , un entier s défini par :

$$s = k + cx$$

- f) l'entité A transmet à l'entité B le message M, la fonction de hachage c et l'entier s précédemment obtenu,
- 5 g) l'entité B vérifie le certificat de la clé publique de A et calcule, à partir du module n, de la clé publique y de l'entité A et de la fonction de hachage c reçue de A, le produit, modulo n, de y^c par g^s , soit :
- $$r = y^c g^s \text{ modulo } n$$
- 10 h) l'entité B vérifie alors que la fonction de hachage du produit r et du message M reçu, soit $h(r, M)$, redonne bien la fonction de hachage c qu'elle a reçue de A, auquel cas l'authentification de la signature du message
- 15 émis par A est réussie.

2. Procédé de signature numérique de messages selon la revendication 1, caractérisé par le fait que la clé publique y est égale à g^{-x} modulo n.

20

3. Procédé de signature numérique de messages selon la revendication 1, dans lequel l'entité A possède un ensemble prédéterminé de valeurs $k_1, \dots, k_i, \dots, k_m$ et un ensemble correspondant d'entiers $r_1, \dots, r_i, \dots, r_m$ liés aux k_i par $r_i = g^{k_i}$ modulo n, l'entité A, pour déterminer ledit entier r, choisissant un nombre k_i dans l'ensemble et lisant l'entier r_i correspondant.

30

4. Procédé selon la revendication 3, caractérisé par le fait que :

- l'autorité et l'entité A possèdent deux générateurs pseudo-aléatoires identiques aptes à délivrer deux suites pseudo-aléatoires d'entiers $k_1, \dots, k_i, \dots, k_m$
- 35 identiques à partir d'une valeur d'initialisation k_0 du

générateur, cette valeur d'initialisation k_0 étant échangée entre l'entité A et l'autorité de manière sécurisée,

- l'autorité calcule la séquence d'entiers $r_i = g^{k_i}$ modulo n correspondant à la séquence $k_1, \dots, k_i, \dots, k_m$ liée à l'initialisation k_0 , transmet la séquence des entiers $r_1, \dots, r_i, \dots, r_m$ à l'entité A qui les mémorise,
- l'entité A devant émettre et signer des messages $M_1, \dots, M_i, \dots, M_m$ met en route son générateur pseudo-aléatoire à l'aide de la valeur d'initialisation k_0 , et pour chaque message M_i à signer lit la valeur k_i délivrée par le générateur pseudo-aléatoire, lit l'entier r_i correspondant et signe le message M_i à l'aide de l'entier r_i .

5. Procédé selon la revendication 4, dans lequel, après avoir signé m messages $M_1, \dots, M_i, \dots, M_m$ et épuisé ses m entiers $r_1, \dots, r_i, \dots, r_m$, l'entité A et l'autorité échangent de manière sécurisée une nouvelle valeur d'initialisation k'_0 pour les deux générateurs pseudo-aléatoires, l'autorité transmet à l'entité A une nouvelle séquence d'entiers $r'_1, \dots, r'_i, \dots, r'_m$, que l'entité A utilise pour une nouvelle série de signature de m messages.

6. Procédé selon la revendication 1, caractérisé par le fait que l'entité A vérifie que $c(t-1) \leq s \leq \text{tuv}-1$ et que si ce n'est pas le cas, l'entité A recommence le processus de signature.

7. Procédé selon la revendication 1, caractérisé par le fait que dans le calcul de la fonction hachage $c = h(r, M)$, le message M est remplacé par une fonction $f(M)$ où f est également une fonction de hachage.

8. Procédé d'authentification de message selon la revendication 1, caractérisé par le fait qu'il comprend d'abord les opérations a, b, c ; puis l'entité A

5 calcule $r' = h(r, M)$; l'entité B choisit ensuite au hasard un entier positif ou nul c inférieur à $v-1$ et l'envoie à l'entité A ; l'entité A calcule ensuite l'entier $s = k + cx$ et envoie à l'entité B le message M et l'entier s ; l'entité B calcule le produit modulo n

10 de y^c par g^s , soit r , et vérifie que $h(r, M) = r'$.

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	
D,A	JOURNAL OF CRYPTOLOGY, vol.4, 1991, USA pages 161 - 174 C.P.SCHNORR 'EFFICIENT SIGNATURE GENERATION BY SMART CARDS' * page 161, ligne 7 - page 163, ligne 42 * * page 166, ligne 16 - page 167, ligne 4 * * page 172, ligne 7 - ligne 31 * * figure 2 * -----	1,3
		DOMAINES TECHNIQUES RECHERCHEES (Int.Cl.5)
		H04L
Date d'achèvement de la recherche		Examineur
28 Octobre 1994		Lydon, M
CATEGORIE DES DOCUMENTS CITES		
X : particulièrement pertinent à lui seul V : particulièrement pertinent en combinaison avec un autre document de la même catégorie A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général O : divulgation non-écrite P : document intercalaire		
T : théorie ou principe à la base de l'invention E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure. D : cité dans la demande I : cité pour d'autres raisons & : membre de la même famille, document correspondant		